

### **REMARKS**

Claims 1, 3-11, and 13-21 are rejected in the application. Claims 1, 10, 11 and 20 have been amended. Claims 2 and 12 were previously canceled. Claims 1, 3-11, and 13-21 remain in the application.

#### **Double Patenting**

Claims 1, 3-11 and 13-21 were rejected under the judicially created doctrine of double patenting over claims 1-17 of U.S. Patent No. 6,658,447. A terminal disclaimer has been submitted in response. Accordingly, withdrawal of the rejection is respectfully requested.

#### **Claim Rejections under 35 U.S.C. § 112, Second Paragraph**

Claims 1, 3-11 and 13-21 were rejected under 35 U.S.C. § 112, second paragraph as failing to distinctly claim the invention. Claims 1, 10, 11, and 20 have been amended to more distinctly claim the invention. Accordingly, withdrawal of the rejection of the claims under 35 U.S.C. § 112, second paragraph is respectfully requested.

#### **Claim Rejections under 35 U.S.C. § 103**

Claims 1, 3-11 and 13-20 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Kimura et al., U.S. Patent No. 6,105,127 (hereinafter “Kimura”) in view of Hewitt et al., U.S. Patent No. 6,339,808 (hereinafter “Hewitt”) in view of Borkenhagen et al., U.S. Patent No. 6,567,839, (hereinafter “Borkenhagen”).

As recited in independent claims 1, 10, 11, and 20, a counter is provided that has a value selected depending on the priority assigned to the thread. Allocating a resource between a

plurality of threads is controlled based on the counter (see pages 7-8 of the filed application). In one embodiment, the counter value can be set high when a high priority is given to a thread, and set to a low value when a low priority is given to a thread. The cited references fail to teach or suggest this feature of the independent claims.

As stated in the Office Action, Kimura fails to disclose a counter to be used in controlling allocation of a resource between a plurality of threads. Hewitt does not either. The cited sections of Hewitt in the Office Action state that the Programmable Interrupt Controller (PIC) includes a timer. There is no disclosure whatsoever in Hewitt for using a counter as described in each of the pending independent claims. The cited table indicates a task priority registers and other priority registers, but the providing of such a list is insufficient for providing the necessary teaching of how the counter is used with respect to such registers as required for a proper § 103 rejection.

Borkenhagen fails to make up for the deficiencies of the Kimura and Hewitt references. The Office Action states that “[Borkenhagen] discloses a thread switch control technique having a counter with the predetermined value for plurality of threads depending on the priority assigned to each thread [col. 5, lines 32 – col. 6, line 11].” A review of this text shows that there is no teaching or suggestion that a counter-value is set depending on a priority of the thread as recited in the claims.

In the paragraph beginning at Col. 5, lines 32, Borkenhagen discloses a thread switch counter. By switching priority between threads, “thrashing” may occur. When this does occur, thread priority is repeatedly being switched between thread without any instructions actually being executed. To combat this, the thread switch counter counts the number of times control switches between threads without an instruction having been executed. When the threshold is

exceeded (e.g., say after 10 such switches), then thread switching is prevented to enable instruction execution.

In the paragraph beginning at Col. 5, line 50, a feature is described that makes sure an excessive amount of time of thread inactivity is avoided. Thus, if thread 0 is inactive for 100 clock cycles, the thread switch counter would initiate a forced thread switch to thread 1 (for example).

In the paragraph beginning at Col. 5, line 62, hardware registers are provided for storing thread states, thread priority, and thread switch conditions.

In the final paragraph of this section, beginning at Col. 5, line 66, it states that the thread priority for a thread may be altered by processing an interrupt request or software instruction. By forcing a change in thread priority, the higher priority thread would be given greater access to processing cycles.

In none of the cited sections, does Borkenhagen teach or suggest that a counter value is selected based on the priority level for the thread. Instead, Borkenhagen describes other aspects of thread priority (e.g., avoiding excessive thread switching). Though Borkenhagen describes providing a counter for thread inactivity, there is no suggestion that the counter value for such a counter is selected based on the priority assigned to the thread as called for in each of the pending claims.

Based on the arguments above, reconsideration and withdrawal of the rejection of claims 1, 3-11, and 13-21 under 35 U.S.C. § 103(a) is respectfully requested.

### CONCLUSION

For all the above reasons, the Applicants respectfully submit that this application is in condition for allowance. A Notice of Allowance is earnestly solicited.

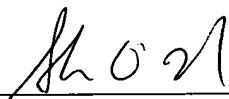
The Examiner is invited to contact the undersigned at (202) 220-4255 to discuss any matter concerning this application. The Office is hereby authorized to charge any additional fees or credit any overpayments under 37 C.F.R. §1.16 or §1.17 to Deposit Account No. **11-0600**.

Respectfully submitted,

KENYON & KENYON

Dated: October 28, 2005

By: \_\_\_\_\_

  
Shawn W. O'Dowd  
Reg. No. 34,687

KENYON & KENYON  
1500 K Street, NW, Suite 700  
Washington, D.C. 20005-1257  
(202) 220-4200 telephone  
(202) 220-4201 facsimile  
DC1-588400v1